UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/735,899	12/14/2000	Stephan J. Jourdan	2207/9807	5754

23838 7590 03/18/2004

KENYON & KENYON
1500 K STREET, N.W., SUITE 700
WASHINGTON, DC 20005

EXAMINER	
O BRIEN, BARRY J	
ART UNIT	**PAPER NUMBER**
2183	

DATE MAILED: 03/18/2004

6

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/735,899	JOURDAN ET AL.	
	Examiner	Art Unit	
	Barry J. O'Brien	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 January 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-19,21-23,28 and 30-36* is/are rejected.

7)☒ Claim(s) *20, 24-27 and 29* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-36 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed on record in the file: Amendment A as received on 1/05/2004.

Specification

3. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

4. The applicant is requested to review the specification and update the status of all co-pending applications made mention of, replacing attorney docket numbers with current U.S. application or patent numbers when appropriate.

Response to Arguments

5. Applicant's arguments, see pages 8-11 of the current Amendment, filed 1/05/2004, with respect to the rejection(s)of claim(s) 1-22 under the Kranich et al. reference have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of Black et al., *The Block-based Trace Cache.*

Claim Rejections - 35 USC § 112

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. Claims 2-5, 7-10, 17 and 36 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

8. Regarding claims 2-5 and 7-10, the claims recite the limitation, "wherein the filtering condition may be met only if." This language is indefinite, as it is unclear if the filtering condition has to be met in order for the segment to be stored in the segment cache. The examiner suggests that the claim language be amended to read, "wherein the filtering condition is met when," in order to more distinctly point out the metes and bounds of the claim. Doing so, along with writing these claims in independent form so as to include the limitations of their parent claims, will overcome the prior art of record (see paragraph 38 below).

9. Regarding claims 17 and 36, the claims recite two different sets of "location flags" on their second and third lines, respectively. It is unclear whether the set of location flags that the segment builder is to use (on the second line of the claim) are the same as the set that are output from the instruction cache system (on the third line of the claim). For the purposes of this examination, the examiner will assume that the first set of location flags that are output from the instruction cache system are not the same as those which are used in determining whether the filtering condition is met, nor do they server any claimed purpose, and thus the claim will be rejected with prior art of record (see paragraphs 20 and 31 below). If the applicant amends the

claim language to read, "whether the filtering condition is met based on said location flags", it would overcome the prior art of record.

Claim Rejections - 35 USC § 102

10. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

11. Claims 1, 6, 11-19, 23, 28 and 30-36 rejected under 35 U.S.C. 102(b) as being anticipated by Black et al., *The Block-based Trace Cache.*

12. Regarding claim 1, Black has taught an instruction segment storing method, comprising:

    a. Building an instruction segment (see Sec. 3.2.2 lines 1-12),

    b. Determining whether the instruction segment satisfies a filtering condition (see Sec. 3.2.2 lines 13-18),

    c. If the instruction segment satisfies the filtering condition, storing the instruction segment in a segment cache (see Sec. 3.2.2 lines 13-18). Here, the filtering condition is whether the block cache already contains the block that the fill unit is attempting to add to the block cache.

13. Regarding claim 6, Black has taught an instruction segment storing method, comprising:

    a. Building an instruction segment (see Sec. 3.2.2 lines 1-12),

    b. Determining, from location flags associated with instructions in the instruction segment, whether the instruction segment satisfies a filtering condition (see Sec.

3.2.2 lines 13-18). Here, there is inherently a signal present to tell the fill unit that the filtering condition has not been meant, namely that a block has already been cached in the block cache (see Sec. 3.2.2 lines 13-18). Because flag is defined as "a variable in a program to inform the program later that a condition has been met" (see Newton's Telecom Dictionary, 18th Ed. p.301 as submitted by the applicant), this inherent signal is a flag, and further it is a location flag as the signal is associated with the location of data in the block cache.

c. If so, storing the instruction segment in a segment cache (see Sec. 3.2.2 lines 13-18).

14. Regarding claim 11, Black has taught a front end system for a processing agent, comprising:

a. An instruction cache system (see I-Cache of Fig.2),

b. An instruction segment system, comprising:

I. A segment cache (see Block Cache of Fig.2),

II. A segment builder (see Fill Unit of Fig.2) provided in communication with the instruction cache system (see Fig.2), to store a new instruction segment in the segment cache when a filtering condition is met (see Sec. 3.2.2 lines 13-18). Here, the filtering condition is whether the block cache already contains the block that the fill unit is attempting to add to the block cache.

15. Regarding claim 12, Black has taught the front-end system of claim 11, further comprising a history map provided in communication with the segment builder (see Rename

Table and Fill Unit of Fig.2) to identify when the filtering condition is met (see Fig.5 and Sec. 3.3 lines 1-10). Here, because the filtering condition is whether the block cache already contains the block that the fill unit would like to store in the block cache (see above paragraph 14), it is inherent that the table which keeps track of which blocks are in the block cache (see Sec. 3.3) will identify if the filtering condition is met or not.

16. Regarding claim 13, Black has taught the front-end system of claim 12, wherein the history map is a direct mapped cache (see Fig.5 and Sec. 3.3 lines 1-15). Here, because the rename table may implement any associativity scheme (see Sec. 3.3 lines 11-15), and because direct mapped is defined as one-way set associativity, Black has inherently taught the rename table being direct mapped.

17. Regarding claim 14, Black has taught the front-end system of claim 12, wherein the history map is a set associative cache (see Fig.5 and Sec. 3.3 lines 1-10).

18. Regarding claim 15, Black has taught the front-end system of claim 14, wherein the history map comprises a plurality of cache entries having a width corresponding to a width of a tag address of an instruction pointer in the system (see Fig.5 and Sec. 3.3). Here, because the hit comparison compares the tag of the instruction pointer with the tag in the rename table, the width of a rename table entry corresponds to the width of the instruction pointer tag.

19. Regarding claim 16, Black has taught the front-end system of claim 14, wherein the history map comprises a plurality of cache entries having a width corresponding to a width of a portion of a tag address of an instruction pointer in the system (see Fig.5 and Sec. 3.3). Here, the width of a rename table entry includes a portion of a tag address, and thus corresponds to a portion of a tag address of an instruction pointer (see Fig.5).

20. Regarding claim 17, Black has taught the front-end system of claim 11, wherein the instruction cache system outputs instructions and location flags to the segment builder, the segment builder determining whether the filtering condition is met based on location flags associated with instructions of the new instruction segment (see Sec. 3.2.2 lines 13-18). Here, there is inherently a signal present to tell the fill unit that the filtering condition has not been meant, namely that a block has already been cached in the block cache (see Sec. 3.2.2 lines 13-18). Because flag is defined as "a variable in a program to inform the program later that a condition has been met" (see Newton's Telecom Dictionary, 18th Ed. p.301 as submitted by the applicant), this inherent signal is a flag, and further it is a location flag as the signal is associated with the location of data in the block cache (see above paragraph 9).

21. Regarding claim 18, Black has taught a processing agent, comprising:

a. A cache hierarchy (see Fetch Buffer and I-Cache of Fig.2),

b. A front end system comprising:

I. An instruction cache system in communication with the cache hierarchy (see I-Cache of Fig.2),

II. An instruction segment system, comprising:

(1) A segment cache (see Block Cache of Fig.2),

(2) A segment builder (see Fill Unit of Fig.2) provided in communication with the instruction cache system (see Fig.2), to store a new instruction segment in the segment cache when a filtering condition is met (see Sec. 3.2.2 lines 13-18). Here, the

filtering condition is whether the block cache already contains the block that the fill unit is attempting to add to the block cache.

22. Regarding claim 19, Black has taught the processing agent of claim 18, further comprising a history map provided in communication with the segment builder (see Rename Table and Fill Unit of Fig.2) to identify when the filtering condition is met (see Fig.5 and Sec. 3.3 lines 1-10). Here, because the filtering condition is whether the block cache already contains the block that the fill unit would like to store in the block cache (see above paragraph 21), it is inherent that the table which keeps track of which blocks are in the block cache (see Sec. 3.3) will identify if the filtering condition is met or not.

23. Regarding claim 23, Black has taught a method comprising:

a. Building an instruction segment (see Sec. 3.2.2 lines 1-12),

b. Determining whether the instruction segment satisfies a filtering condition (see Sec. 3.2.2 lines 13-18),

c. Storing the instruction segment in a segment cache unless the instruction segment does not satisfy the filtering condition (see Sec. 3.2.2 lines 13-18). Here, the filtering condition is whether the block cache already contains the block that the fill unit is attempting to add to the block cache.

24. Regarding claim 28, Black has taught the method of claim 23, wherein the determining comprises determining whether the instruction segment has been built at least twice (see Sec. 3.2.2 lines 13-18).

25. Regarding claim 30, Black has taught a front end system for a processing agent, comprising:

a. An instruction cache system (see I-Cache of Fig.2),

b. An instruction segment system, comprising:

I. A segment builder (see Fill Unit of Fig.2) to build instruction segments from instructions retrieved from the instruction cache system (see Fig.2 and Sec. 3.2.2 lines 1-12),

II. A segment cache (see Block Cache of Fig.2) to store instruction segments unless the instruction segments fail a filtering condition (see Sec. 3.2.2 lines 13-18). Here, the filtering condition is whether the block cache already contains the block that the fill unit is attempting to add to the block cache.

26. Regarding claim 31, Black has taught the front end system of claim 30, further comprising a history map provided in communication with the segment builder (see Rename Table and Fill Unit of Fig.2) to identify when the filtering condition is met (see Fig.5 and Sec. 3.3 lines 1-10). Here, because the filtering condition is whether the block cache already contains the block that the fill unit would like to store in the block cache (see above paragraph 25), it is inherent that the table which keeps track of which blocks are in the block cache (see Sec. 3.3) will identify if the filtering condition is met or not.

27. Regarding claim 32, Black has taught the front-end system of claim 31, wherein the history map is a direct mapped cache (see Fig.5 and Sec. 3.3 lines 1-15). Here, because the rename table may implement any associativity scheme (see Sec. 3.3 lines 11-15), and because direct mapped is defined as one-way set associativity, Black has inherently taught the rename table being direct mapped.

28. Regarding claim 33, Black has taught the front-end system of claim 31, wherein the history map is a set associative cache (see Fig.5 and Sec. 3.3 lines 1-10).

29. Regarding claim 34, Black has taught the front-end system of claim 33, wherein the history map comprises a plurality of cache entries having a width corresponding to a width of a tag address of an instruction pointer in the system (see Fig.5 and Sec. 3.3). Here, because the hit comparison compares the tag of the instruction pointer with the tag in the rename table, the width of a rename table entry corresponds to the width of the instruction pointer tag.

30. Regarding claim 35, Black has taught the front-end system of claim 33, wherein the history map comprises a plurality of cache entries having a width corresponding to a width of a portion of a tag address of an instruction pointer in the system (see Fig.5 and Sec. 3.3). Here, the width of a rename table entry includes a portion of a tag address, and thus corresponds to a portion of a tag address of an instruction pointer (see Fig.5).

31. Regarding claim 36, Black has taught the front-end system of claim 30, wherein the instruction cache system outputs instructions and location flags to the segment builder, the segment builder determining whether the filtering condition is met based on location flags associated with instructions of the new instruction segment (see Sec. 3.2.2 lines 13-18). Here, there is inherently a signal present to tell the fill unit that the filtering condition has not been meant, namely that a block has already been cached in the block cache (see Sec. 3.2.2 lines 13-18). Because flag is defined as "a variable in a program to inform the program later that a condition has been met" (see Newton's Telecom Dictionary, 18th Ed. p.301 as submitted by the applicant), this inherent signal is a flag, and further it is a location flag as the signal is associated with the location of data in the block cache (see above paragraph 9).

Claim Rejections - 35 USC § 103

32. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

33. Claims 21 and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Black et al., *The Block-based Trace Cache.*

34. Regarding claim 21, Black has taught a computer system, comprising the processing agent of claim 18 as shown above, wherein the cache hierarchy includes an internal cache (see I-Cache of Fig.2), but has not explicitly taught wherein the cache hierarchy includes a system memory.

35. However, Official Notice is taken that modern microprocessors include a system memory, which is larger yet slower than an instruction cache, to store non-trivial program instructions in so that the appearance of a large yet fast memory structure is created. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Black to include a system memory so as to provide an large storage area that combined with the instruction cache provides the appearance of a large, fast memory structure.

36. Regarding claim 22, Black has taught a computer system, comprising the processing agent of claim 18 as shown above, wherein the cache hierarchy includes an internal cache (see I-Cache of Fig.2), but has not explicitly taught wherein the cache hierarchy includes an external cache.

37. However, Official Notice is taken that modern microprocessors include a memory hierarchy consisting of a system memory and a L1 or L2 cache, which are incrementally larger yet slower than an instruction cache, to store non-trivial program instructions in so that the appearance of a large yet fast memory structure is created. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Black to include an external cache so as to provide an larger storage area that, combined with the instruction cache, provides the appearance of a large, fast memory structure to store non-trivial program instructions in.

Allowable Subject Matter

38. Claims 2-5, 7-10, 20, 24-27, and 29 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims. The prior art of record has taught the limitations of parent claims 1, 6, 18 and 23, namely an instruction segment storing method that builds a segment, yet only stores the segment in the segment cache if a certain filtering condition has been met, wherein the filtering condition is whether the segment already exists in the segment cache. However, the prior art of record has not taught the specific filtering conditions as specified in these objected claims, specifically that segments can only be stored in the segment cache if some portion of the instructions in the segment came from the instruction cache.

Conclusion

39. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the

patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

40. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Barry J. O'Brien whose telephone number is (703) 305-5864. The examiner can normally be reached on Mon.-Fri. 7:00am-4:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

41. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Barry J. O'Brien
Examiner
Art Unit 2183

BJO
3/16/2004

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100